BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

REMARKS

Reconsideration of the present application is respectfully requested in view of the following remarks. Prior to entry of this response, Claims 1-2, 4-7, 9, 11-12, and 14-20 were pending in the application, of which Claims 1, 7, 9, 11, and 12 are independent. In the Final Office Action dated October 22, 2003, all pending claims were rejected under 35 U.S.C. §102(b). Following this response, Claims 1-2, 4-7, 9, 11-12, and 14-20 remain in this application. Applicant hereby addresses the Examiner's rejections in turn.

I. Rejection of the Claims Under 35 U.S.C. § 102(b)

In the Final Office Action dated October 22, 2003, the Examiner rejected Claims 1-2, 4-7, 9, 11-12, and 14-20 under 35 U.S.C. § 102(b) as being anticipated by U.S. Patent No. 5,668,998 ("*Mason*"). Claims 1, 7, 9, 11, and 12 have been amended to further define and clarify the invention, and Applicant respectfully submits that the amendment overcomes this rejection and adds no new matter.

Amended Claim 1 is patentably distinguishable over the cited art in that it recites, for example, the user interface part being placed on a boundary between a computer and the user. Amended Claims 7, 9, 11, and 12 include similar recitations.

In the Final Office Action, the Examiner states that the API in FIG. 2 of *Mason* discloses a user interface part for receiving instructions from a user and for presenting data to the user when the user employs the constructed service providing system, as recited in independent Claims 1, 7, 9, and 12. Furthermore, in the Advisory Action, the Examiner states that as previously pointed out in Paper No. 9, *Mason* teaches a user interface part for receiving instructions from a user and for presenting data to the user

when the user employs the constructed service providing system (column 7, lines 64-67 to column 8, lines 1-3; column 8, lines 56-67 to column 9, lines 1-18; for example, see Fig. 2, 'API", on the SCU side). In addition, the Examiner states that *Mason's* API on the SCU side is interpreted as the user interface part where the application developer chooses the DT Service Interface object to initiate a request via the API. Furthermore, according to the Examiner, the application developer in *Mason* receives the status/confirmation of the request. If the status was not a success in *Mason*, then the application developer, according to the Examiner, needs to recover from this status, for example, by creating a subclass of the appropriate DT Service Interface class.

While FIG. 2 may disclose an API, nowhere in *Mason* does it disclose the API receiving and presenting data when the user employs the constructed service providing system. Rather, *Mason's* API is merely an interface through which an application programmer customizes individual objects in the framework or alters parameter values and object behavior when developing the framework. *Mason's* API does not receive instructions from a user or present data to the user when the user employs the constructed service providing system. Accordingly, *Mason's* API toolkit framework does not disclose the user interface part, as recited in independent Claims 1, 7, 9, and 12, which receives instructions from a user and presents data to the user when the user employs the constructed service providing system.

Furthermore, *Mason's* Fig. 2 merely discloses that the interface (e.g., the interface for encoding / decoding image data) used by the handler objects (SCP/SCU) may be defined as an Application Programmers Interface (API). This does not mean that *Mason*'s API is an interface placed on a boundary between a computer node and

an end user (e.g., a person who employs the computer node). Instead, *Mason*'s API corresponds to an interface placed between a first computer node and a second computer node so that the first computer may communicate image data with the second computer node through the API.

In contrast, however, the claimed user interface part, for example, may be placed on a boundary between a computer node and an end user (a person, for example, who employs the computer node). This means that the claimed user interface part may, for example, be an API for user-to-node, and may not be an API for node-to-node.

Moreover, in *Mason*'s Fig. 2, the term "User" in "Service Class User (SCU)" is merely used to contrast it with the term "Provider" in "Service Class Provider (SCP)." Namely, both the SCU and the SCP are computer nodes, one of which (e.g., the SCU) receives image data from the other computer node (e.g., the SCP), whereas the other of which (e.g., the SCP) sends image data to the one computer node (e.g., the SCU). In other words, the terms "User" and "Provider" merely represent the roles or behaviors of the computer nodes connected via a network, and they may not represent end users who employ the medical system constructed by the framework. In addition, the SCU is incorporated into a middleware for communication, and may not exist inside an application. (*See Mason*, col. 8, lines 56-67 and col. 9, lines 1-18.)

In contrast, the claimed user, for example, may mean an end user who actually employs the service providing system constructed by the framework. Moreover, the claimed user interface part may, for example, receive instructions from an end user or may display the status of the object system.

In summary, *Mason's* SCU merely decodes image data sent from the SCP and transmits a message to the SCU. *Mason's* SCU does not include a user interface that is placed on a boundary between a computer node and an end user.

Furthermore, *Mason's* API toolkit framework merely comprises an interface through which an application programmer may select to create an application that provides a particular DICOM service. (*See Mason*, col. 3, lines 16-21.) *Mason's* API toolkit framework has nothing to do with either the SCU (as an API for node-to-node) or the claimed user interface part (for example, an API for user-to-node).

In addition, the Examiner states in the Final Office Action that the handler object (SCP/SCU) of *Mason* discloses an integrated control part for controlling said data holding part, said user interface part and said object system interface part, as recited in independent Claims 1, 7, 9, and 12, and for controlling said internal system means and said object system interface means, as recited in independent Claim 11. *Mason*, however, merely discloses that the SCP/SCU ensures that messages and events are in appropriate DICOM standard format. (*See* col. 2, lines 41-43.) *Mason* further discloses that the SCP/SCU enables an application to send and return calls from other applications. (*See* col. 2, lines 44-46.) Unlike the claimed integrated control part, *Mason's* SCP/SCU does not control anything, much less a data holding part, a user interface part, or an object system interface part, as recited in Claims 1, 7, 9, and 12, or an internal system means or an object system interface means, as recited in Claim 11.

Furthermore, *Mason's* handler objects (SCP/ SCU) merely enable an application to send and return calls to and from other applications. (*See* col. 2, lines 43-44.) This means that *Mason's* handler objects (SCP/ SCU) are incorporated into a middleware for communication, and thus do not exist inside an application.

Moreover, *Mason's* actual framework structure is completely different from the Examiner's assertions that the handler objects (SCP/SCU) are an integrated control part for controlling the DICOM service collection of objects. For example, although the Examiner asserts that the DICOM service collection of objects corresponds to the claimed holding part, the DICOM service collection of objects may simply correspond to the handler objects (SCP/SCU) that accomplish DICOM services.

Also, although the Examiner asserts that the API corresponds to the claimed user interface part, *Mason's* API is merely an interface placed between one computer node and the other computer node, as described above. Accordingly, *Mason's* API has nothing to do with the claimed user interface part, which may be placed on a boundary between a computer node and an end user.

In addition, although the Examiner asserts that the application interface corresponds to the claimed object system interface part, the term "application interface" may merely be a general expression of "API." According to *Mason's* Fig. 2, there is no component called "application interface."

In this regard, even if the DICOM service collection of objects, API, and application interface are interpreted as a data holding part, a user interface part, and an object system interface part, respectively, the claimed invention is completely different from *Mason*. For example, although the claimed invention, for example, may include

that the user interface part "utilizes" the data holding part, *Mason* does not disclose the API utilizing the DICOM service collection of objects. In addition, although the claimed invention may include, for example, that the integrated control part and the object system interface part utilize each other, *Mason* does not disclose that the handler objects (SCP/SCU) and the application interface utilize each other.

In short, *Mason* does not disclose the above referenced recitations of independent Claims 1, 7, 9, 11, and 12. Accordingly, independent Claims 1, 7, 9, 11, and 12 patentably distinguish the present invention over the cited art, and Applicant respectfully requests withdrawal of the rejection of Claims 1, 7, 9, 11, and 12.

Dependent Claims 2, 4-6, and 14-20 are also allowable at least for the reasons above regarding independent Claims 1 and 12 and by virtue of their respective dependencies upon independent Claims 1 and 12. Accordingly, Applicant respectfully requests withdrawal of this rejection of dependent Claims 2, 4-6, and 14-20.

II. Conclusion

In view of the foregoing remarks, Applicant respectfully submits that the claimed invention, as amended, is neither anticipated nor rendered obvious in view of the prior art references cited against this application. Applicant therefore requests the entry of this Amendment, the Examiner's reconsideration and reexamination of the application, and the timely allowance of the pending claims.

In view of the foregoing, Applicant respectfully submits that the pending claims, as amended, are patentable over the cited references. The preceding arguments are based only on the arguments in the Official Action, and therefore do not address patentable aspects of the invention that were not addressed by the Examiner in the Official Action. The claims may include other elements that are not shown, taught, or suggested by the cited art. Accordingly, the preceding argument in favor of patentability is advanced without prejudice to other bases of patentability.

Please grant any extensions of time required to enter this amendment and charge any additional required fees to our Deposit Account No. 06-0916.

Respectfully submitted,

FINNEGAN, HENDERSON, FARABOW,
GARRETT & DUNNER, L.L.P.

Dated: 4/21/04

By: _____
Richard V. Burgujian
Registration No. 31,744